

AUS920030606US1

RECEIVED  
CENTRAL FAX CENTER

JUN 29 2006

## AMENDMENT

Please amend the above-identified application as follows:

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the above-identified application:

Listing of Claims:

1. (Currently Amended) A method of content addressable data storage and compression for computer memory comprising:

~~providing a chunk of data comprising a quantity of input data;~~

~~retrieving a memory block from computer memory;~~

~~searching for a segment of the chunk that matches the memory block; and~~

searching at a repeating memory interval through a search section of a chunk for a segment of the chunk that matches a memory block from computer memory, including: calculating a weak checksum for the memory block; calculating rolling weak checksums for segments of the search section of the chunk; comparing the rolling weak checksums for the segments with the checksum for the memory block; and if a segment is found with a rolling weak checksum equal to the weak checksum of the memory block: calculating a strong checksum for the memory block; calculating a strong checksum for the segment with the matching rolling weak checksum; comparing the strong checksum of the memory block and the strong checksum for the segment with the equal rolling weak checksum;

AUS920030606US1

determining that the search has found a segment having contents that match the contents of the memory block if the strong checksum of the memory block and the strong checksum for the segment with the matching rolling weak checksum are equal;

if a matching segment is found:

discarding the matching segment and providing a retrieval key for the memory block as a retrieval key for the matching segment;

identifying an unmatched portion of the chunk that does not match the memory block; and

storing the unmatched portion and providing a retrieval key for the unmatched portion.

2. (Original) The method of claim 1 wherein providing a chunk comprises retrieving from input a quantity of input data for storage and compression having a chunk size larger than a maximum memory block size.
3. (Original) The method of claim 1 wherein retrieving a memory block from computer memory comprises retrieving from computer memory a memory block having a memory block size no greater than a maximum memory block size.
4. (Cancelled)
5. (Currently Amended) The method of claim 14 wherein the memory interval is one bit.
6. (Cancelled)

AUS920030606US1

7. (Currently Amended) The method of claim 16 wherein calculating a strong checksum for the memory block comprises calculating a static strong checksum for the memory block.
8. (Cancelled)
9. (Original) The method of claim 1 wherein storing the unmatched portion of the chunk comprises storing the unmatched portion of the chunk as a new memory block having a memory block size equal to the size of the unmatched portion of the chunk.
10. (Original) The method of claim 1 wherein providing a retrieval key for the unmatched portion of a chunk comprises:  
  
calculating a weak checksum for the unmatched portion of the chunk; and  
  
calculating a strong checksum for the unmatched portion of the chunk.
11. (Current Amended) The method of claim 1 wherein further comprising providing at the chunk, including ~~comprises~~:  
  
retrieving from input a quantity of data equal in size to the sum of the sizes of the matching segment and the unmatched portion; and  
  
concatenating the retrieved input quantity to a remaining portion of the chunk that remains after discarding the matching segment and storing the unmatched portion.
12. (Original) The method of claim 1 wherein searching for a segment of the chunk that matches the memory block fails to find a matching segment, the method further comprising repeatedly carrying out the following steps for all memory blocks in computer memory until a matching segment is found:

AUS920030606US1

retrieving a next memory block from computer memory; and

searching for a segment of the chunk that matches the next memory block.

13. (Original) The method of claim 12 wherein no matching segment is found in any memory block in computer memory, the method further comprising:

storing a search section of the chunk; and

providing a retrieval key for the search section of the chunk.

14. (Original) The method of claim 13 wherein storing a search section of the chunk comprises storing the search section of the chunk as a new memory block having a memory block size equal to the size of the search section of the chunk.

15. (Original) The method of claim 13 wherein providing a retrieval key for a search section of a chunk comprises:

calculating a weak checksum for the search section of the chunk; and

calculating a strong checksum for the search section of the chunk.

16. (Currently Amended) The method of claim 13 ~~wherein further comprising~~ providing ~~at the chunk, including comprises:~~

retrieving from input a quantity of data equal in size to the search section; and

concatenating the retrieved input quantity to the remaining portion of the chunk that remains after storing the search section.

AUS920030606US1

17. (Original) The method of claim 1 further comprising:

receiving a retrieval key;

identifying a memory block in dependence upon the retrieval key; and

retrieving the identified memory block.

18. (Currently Amended) A system of content addressable data storage and compression for computer memory comprising:

~~means for providing a chunk of data comprising a quantity of input data;~~

~~means for retrieving a memory block from computer memory;~~

~~means for searching for a segment of the chunk that matches the memory block;~~

means for searching at a repeating memory interval through a search section of a chunk for a segment of the chunk that matches a memory block from computer memory, including means for: calculating a weak checksum for the memory block; calculating rolling weak checksums for segments of the search section of the chunk; comparing the rolling weak checksums for the segments with the checksum for the memory block; and if a segment is found with a rolling weak checksum equal to the weak checksum of the memory block: calculating a strong checksum for the memory block; calculating a strong checksum for the segment with the matching rolling weak checksum; comparing the strong checksum of the memory block and the strong checksum for the segment with the equal rolling weak checksum;

AUS920030606US1

means for determining that the search has found a segment having contents that match the contents of the memory block if the strong checksum of the memory block and the strong checksum for the segment with the matching rolling weak checksum are equal;

means for discarding a matching segment and providing a retrieval key for the memory block as a retrieval key for the matching segment;

means for identifying an unmatched portion of the chunk that does not match the memory block; and

means for storing the unmatched portion and providing a retrieval key for the unmatched portion.

19. (Original) The system of claim 18 wherein means for providing a chunk comprises means for retrieving from input a quantity of input data for storage and compression having a chunk size larger than a maximum memory block size.
20. (Original) The system of claim 18 wherein means for retrieving a memory block from computer memory comprises means for retrieving from computer memory a memory block having a memory block size no greater than a maximum memory block size.
21. (Cancelled)
22. (Currently Amended) The system of claim ~~1824~~ wherein the memory interval is one bit.
23. (Cancelled)
24. (Currently Amended) The system of claim ~~1823~~ wherein means for calculating a

AUS920030606US1

strong checksum for the memory block comprises means for calculating a static strong checksum for the memory block.

25. (Cancelled)

26. (Original) The system of claim 18 wherein means for storing the unmatched portion of the chunk comprises means for storing the unmatched portion of the chunk as a new memory block having a memory block size equal to the size of the unmatched portion of the chunk.

27. (Original) The system of claim 18 wherein means for providing a retrieval key for the unmatched portion of a chunk comprises:

means for calculating a weak checksum for the unmatched portion of the chunk;  
and

means for calculating a strong checksum for the unmatched portion of the chunk.

28. (Original) The system of claim 18 wherein further comprising means for providing a chunk, including comprises:

means for retrieving from input a quantity of data equal in size to the sum of the sizes of the matching segment and the unmatched portion; and

means for concatenating the retrieved input quantity to a remaining portion of the chunk that remains after discarding the matching segment and storing the unmatched portion.

29. (Original) The system of claim 18 wherein means for searching for a segment of the chunk that matches the memory block fails to find a matching segment, the method further comprising:

AUS920030606US1

means for retrieving a next memory block from computer memory; and

means for searching for a segment of the chunk that matches the next memory block.

30. (Currently Amended) The system of claim 1829 further comprising:

means for storing a search section of the chunk; and

means for providing a retrieval key for the search section of the chunk.

31. (Currently Amended) The system of claim 1830 wherein means for storing a search section of the chunk comprises means for storing the search section of the chunk as a new memory block having a memory block size equal to the size of the search section of the chunk.

32. (Currently Amended) The system of claim 1830 wherein means for providing a retrieval key for a search section of a chunk comprises:

means for calculating a weak checksum for the search section of the chunk; and

means for calculating a strong checksum for the search section of the chunk.

33. (Currently Amended) The system of claim 1830 ~~wherein~~ further comprising means for providing a chunk, including ~~comprises~~:

means for retrieving from input a quantity of data equal in size to the search section; and



AUS920030606US1

means for concatenating the retrieved input quantity to the remaining portion of the chunk that remains after storing the search section.

34. (Original) The system of claim 18 further comprising:

means for receiving a retrieval key;

means for identifying a memory block in dependence upon the retrieval key; and

means for retrieving the identified memory block.

35. (Currently Amended) A computer program product of content addressable data storage and compression for computer memory comprising:

a recording medium;

~~means, recorded on the recording medium, for providing a chunk of data comprising a quantity of input data;~~

~~means, recorded on the recording medium, for retrieving a memory block from computer memory;~~

~~means, recorded on the recording medium, for searching for a segment of the chunk that matches the memory block;~~

means, recorded on the recording medium, for searching at a repeating memory interval through a search section of a chunk for a segment of the chunk that matches a memory block from computer memory, including means, recorded on the recording medium, for: calculating a weak checksum for the memory block; calculating rolling weak checksums for segments of the search section of the chunk; comparing the rolling weak checksums for the segments with the

AUS920030606US1

checksum for the memory block; and if a segment is found with a rolling weak checksum equal to the weak checksum of the memory block; calculating a strong checksum for the memory block; calculating a strong checksum for the segment with the matching rolling weak checksum; comparing the strong checksum of the memory block and the strong checksum for the segment with the equal rolling weak checksum;

means, recorded on the recording medium, for determining that the search has found a segment having contents that match the contents of the memory block if the strong checksum of the memory block and the strong checksum for the segment with the matching rolling weak checksum are equal;

means, recorded on the recording medium, for discarding a matching segment and providing a retrieval key for the memory block as a retrieval key for the matching segment;

means, recorded on the recording medium, for identifying an unmatched portion of the chunk that does not match the memory block; and

means, recorded on the recording medium, for storing the unmatched portion and providing a retrieval key for the unmatched portion.

36. (Original) The computer program product of claim 35 wherein means, recorded on the recording medium, for providing a chunk comprises means, recorded on the recording medium, for retrieving from input a quantity of input data for storage and compression having a chunk size larger than a maximum memory block size.
37. (Original) The computer program product of claim 35 wherein means, recorded on the recording medium, for retrieving a memory block from computer memory comprises means, recorded on the recording medium, for retrieving from

AUS920030606US1

computer memory a memory block having a memory block size no greater than a maximum memory block size.

38. (Cancelled)
39. (Currently Amended) The computer program product of claim ~~35~~<sup>38</sup> wherein the memory interval is one bit.
40. (Cancelled)
41. (Currently Amended) The computer program product of claim ~~35~~<sup>40</sup> wherein means, recorded on the recording medium, for calculating a strong checksum for the memory block comprises means, recorded on the recording medium, for calculating a static strong checksum for the memory block.
42. (Cancelled)
43. (Original) The computer program product of claim 35 wherein means, recorded on the recording medium, for storing the unmatched portion of the chunk comprises means, recorded on the recording medium, for storing the unmatched portion of the chunk as a new memory block having a memory block size equal to the size of the unmatched portion of the chunk.
44. (Original) The computer program product of claim 35 wherein means, recorded on the recording medium, for providing a retrieval key for the unmatched portion of a chunk comprises:
- means, recorded on the recording medium, for calculating a weak checksum for the unmatched portion of the chunk; and

AUS920030606US1

means, recorded on the recording medium, for calculating a strong checksum for the unmatched portion of the chunk.

45. (Original) The computer program product of claim 35 ~~wherein~~ further comprising means, recorded on the recording medium, for providing a chunk, including ~~comprises:~~

means, recorded on the recording medium, for retrieving from input a quantity of data equal in size to the sum of the sizes of the matching segment and the unmatched portion; and

means, recorded on the recording medium, for concatenating the retrieved input quantity to a remaining portion of the chunk that remains after discarding the matching segment and storing the unmatched portion.

46. (Original) The computer program product of claim 35 wherein means, recorded on the recording medium, for searching for a segment of the chunk that matches the memory block fails to find a matching segment, the method further comprising:

means, recorded on the recording medium, for retrieving a next memory block from computer memory; and

means, recorded on the recording medium, for searching for a segment of the chunk that matches the next memory block.

47. (Currently Amended) The computer program product of claim 35~~46~~ further comprising:

means, recorded on the recording medium, for storing a search section of the chunk; and

AUS920030606US1

means, recorded on the recording medium, for providing a retrieval key for the search section of the chunk.

48. (Currently Amended) The computer program product of claim 3547 wherein means, recorded on the recording medium, for storing a search section of the chunk comprises means, recorded on the recording medium, for storing the search section of the chunk as a new memory block having a memory block size equal to the size of the search section of the chunk.

49. (Currently Amended) The computer program product of claim 3547 wherein means, recorded on the recording medium, for providing a retrieval key for a search section of a chunk comprises:

means, recorded on the recording medium, for calculating a weak checksum for the search section of the chunk; and

means, recorded on the recording medium, for calculating a strong checksum for the search section of the chunk.

50. (Currently Amended) The computer program product of claim 3547 wherein further comprising means, recorded on the recording medium, for providing a chunk, including ~~comprises~~:

means, recorded on the recording medium, for retrieving from input a quantity of data equal in size to the search section; and

means, recorded on the recording medium, for concatenating the retrieved input quantity to the remaining portion of the chunk that remains after storing the search section.

AUS920030606US1

51. (Original) The computer program product of claim 35 further comprising:

means, recorded on the recording medium, for receiving a retrieval key;

means, recorded on the recording medium, for identifying a memory block in dependence upon the retrieval key; and

means, recorded on the recording medium, for retrieving the identified memory block.